

Nattan MAISONNEUVE–FLEURY
BTS SIO 1
Groupe B

23&30/09/2025

Compte Rendu - Java 4

Sommaire :

Introduction :	1
SHIFUMI ! :	1
Conclusion :	5

Introduction :

Dans ce TP, nous allons voir comment créer le jeu du shifumi en langage Java.

SHIFUMI ! :

Tout d'abord, il faut créer l'environnement de travail. Sur eclipse, appuyez sur File -> New -> JavaProject. Ensuite Mettez un nom au projet, par exemple "Jeu_Shifumi". Finalement appuyez sur Finish.

Sur la gauche, faites cliques droit sur le dossier "Jeu_Shifumi" -> New -> Class. Dans Name marquez par exemple Shifumi. Maintenant cochez la case public static void main (string[] args) Et enfin appuyer sur finish.

Maintenant commençons à coder :

- 1) Il faut commencer par afficher toutes les règles à l'utilisateur. Pour ce faire, nous pouvons tout simplement utilisé les lignes de commandes println pour afficher du texte, comme ceci :

```
System.out.println("Voici les règles du jeu:");  
System.out.println("Feuille bat la pierre");  
System.out.println("Pierre bat les ciseaux");  
System.out.println("Ciseaux bat la feuille");  
System.out.println("égalité si deux élément similaire");
```

- 2) Maintenant nous pouvons ajouter la bibliothèque Scanner pour que l'utilisateur puisse écrire et récupérer les informations pour les stocker dans une variable. Nous pouvons le faire grâce a cette ligne que nous collons a la suite du code :

```
Scanner sc = new Scanner(System.in);
```

Et au tout début du code a la première ligne celle ci :

```
import java.util.Scanner;
```

Etablir les règles du jeu :

- 2 joueurs
- Chacun choisit soit pierre, feuille ou ciseaux
- Le ciseaux bat la feuille
- La pierre bat le ciseaux
- La feuille bat la pierre
- Si les 2 joueurs prennent le même élément il y a égalité

Voici en ligne de code le résultat :

- 1) Demander à l'utilisateur le nombre de points pour gagner

Variables :

```
int nbpoints; // Variables nombre de points
```

Code :

```
do { // répéter....
System.out.println("En combien de points se déroule la partie ? (3, 5 ou 10)"); // affiche la phrase...
nbpoints= sc.nextInt(); // la variable nbpoints prend la valeur que rentre l'utilisateur
System.out.println("Vous avez saisi : "+nbpoints); //affiche la phrase ... + la variable
}while(nbpoints != 3 && nbpoints != 5 && nbpoints != 10); // tant que l'utilisateur n'a pas rentré, 3, 5 ou 10.
```

- 2) Demander à l'utilisateur Pierre, Feuille ou ciseau

Variables :

```
char chjoueur; // Variables choix joueur
```

Code :

```
do { // répéter
System.out.println("Choisissez pierre, feuille, ciseau"); // afficher la phrase...
chjoueur = sc.next().charAt(0); // la variable chjoueur prend la valeur que rentre l'utilisateur (p, c ou f)
System.out.println("Vous avez saisi : "+chjoueur); // affiche la phrase ... + la variable
}while(chjoueur != 'p' && chjoueur != 'c' && chjoueur != 'f'); // tant que l'utilisateur n'a pas rentré, 3, 5 ou 10.
```

3) Choix aléatoire de l'ordinateur

Variables :

```
char chOrdi; // variable choix de l'ordinateur
int aleatoire; // variable nombre aléatoire
```

Code :

```
aleatoire= (int)(Math.random()*3) +1; // nombre aléatoire entre 1 & 3
if (aleatoire==1) { // si variable aléatoire = 1
    chOrdi='X'; // alors variable chOrdi prend valeur X
}
else if (aleatoire==2) { // sinon si variable aléatoire = 2
    chOrdi='_'; // alors variable chOrdi prend valeur _
}
else {
    chOrdi='>'; // sinon variable chOrdi prend valeur >
}
```

4) Révélation du suspense

Code :

```
System.out.println("l'ordinateur a choisi..."); // affiche la phrase ...
//try catch permet de continuer le programme en cas de problème de la fonction thread
try {
    Thread.sleep(3000); // instaure une durée de 3000ms soit 3s
} catch (InterruptedException e) {
    e.printStackTrace(); // fin de la durée
}
System.out.println("fin du suspense !"); // affiche la phrase ...
System.out.println(chOrdi); // affiche la valeur qu'à "choisit" l'ordinateur
```

5) Déterminer le gagnant de la manche et attribuer les points

Variables :

```
int scoreOrdi = 0; // initialise la variable score ordinateur à 0
int scoreJoueur = 0; // initialise la variable score joueur à 0
```

```

System.out.println("Choix de l'ordinateur : " + chOrdi); // affiche la phrase ... + variable
if (chJoueur == chOrdi) { // si variable chJoueur = chOrdi
    System.out.println("Égalité !"); // alors affiché ...
} else if ((chJoueur == 'p' && chOrdi == 'C') || // sinon si chJoueur = p et chOrdi = c
(chJoueur == 'f' && chOrdi == 'P') || // ou chJoueur = f et chOrdi = p
(chJoueur == 'c' && chOrdi == 'F')) { // ou chJoueur = c et chOrdi = f
    System.out.println("Vous gagnez !"); // alors affiché ...
    scoreJoueur += 1; // et la variable scoreJoueur prend +1
} else { // sinon
    System.out.println("L'ordinateur gagne !"); // afficher la phrase ...
    scoreOrdi += 1; // et la variable scoreOrdi prend +1
}

System.out.println("Score : Joueur " + scoreJoueur + " - Score ordi : " + scoreOrdi); // afficher
le score grâce à phrase... + variable...
}

```

6) Boucle des manches

Pour cette étapes il faut se placer juste avant l'étape 2 et coller ceci

```

while (scoreJoueur < nbpoints && scoreOrdi < nbpoints) { // tant que la variable scoreJoueur
et scoreOrdi sont inférieure à la variable nbpoints (3, 5, 10) faire tout le programme choisir un
élément , attendre....

```

à la fin du code il faut aussi placer un }

7) Fin de partie

Encore une fois pour cette partie, il faut revenir au début, juste après les variables et noter :

```

while (rejouer) {

```

Et avant les trois }

```

System.out.println("Voulez-vous rejouer ? (o/n)"); // demande à l'utilisateur de choisir o ou n
char reponse = sc.next().charAt(0); // Prend le premier caractère du prochain mot saisi par
l'utilisateur
rejouer = (reponse == 'o'); // Me rejouert à true si ce caractère est 'o', sinon à false.

```

8) Mise en forme

Pour donner plus de vie à notre jeu, nous pouvons remplacer les lettres par des symboles comme ceci :

```
(chJoueur == '♠' && chOrdi == '>') || // ♠ pour pierre  
(chJoueur == '♣' && chOrdi == '♠') || // ♣ pour feuille  
(chJoueur == '>' && chOrdi == '♣') || // > pour ciseau
```

9) Ajouter le Puit

*voir fichier `Shifumi_puit.java`

10) Choix du mode

Ici, nous pouvons créer une nouvelle page `accueil_shifumi.java` dans laquelle nous allons choisir soit avec puit soit sans puit.

*voir fichier `accueil_shifumi.java`

Dans ce fichier on dit si l'utilisateur choisit "avec puit" alors on exécute la page...
"sinon" on exécute la page...

Conclusion :

Ce TP a permis de nous apprendre de nouvelles instructions en Java et de s'imprégner de celles que nous connaissions déjà. J'ai trouvé ce TP très intéressant malgré quelques étapes que j'ai trouvé compliquées comme la 7 ou la 10.

